An exploratory study on user interaction challenges when handling interconnected requirements artifacts of various sizes

Ghazi, Parisa ; Glinz, Martin

Abstract: Requirements documentation is essential for developing software systems of non-trivial size. The cost of creating and maintaining documentation artifacts in terms of time and effort is significantly influenced by the tools with which engineers view, navigate and edit documentation artifacts. However, there is not much evidence about how well documentation tools actually support engineers, particularly when dealing with artifacts that are larger than the available display screen and with multiple artifacts at the same time. Therefore, we conducted an exploratory study based on 29 interviews with software practitioners to understand the current practice of presenting and manipulating artifacts in documentation tools, and how practitioners deal with the challenges encountered. Our study shows that a significant number of artifacts cannot be viewed entirely, even on large screens. Moreover, more than half of the participants use four or more artifacts concurrently. Nevertheless, current tools only provide primitive capabilities for handling concurrent and large artifacts, thus forcing engineers to create, for example, mental images of the currently used artifacts or use workarounds such as hanging printouts to the wall. Our results may trigger new research and help improve requirements engineering tools.

DOI: https://doi.org/10.1109/RE.2016.52

Originally published at:
Ghazi, Parisa; Glinz, Martin (2016). An exploratory study on user interaction challenges when handling interconnected requirements artifacts of various sizes. In: Proceedings of the 24th IEEE International Requirements Engineering Conference, Beijing, China, 12 September 2016 - 16 September 2016, IEEE.
DOI: https://doi.org/10.1109/RE.2016.52

An Exploratory Study on User Interaction Challenges When Handling Interconnected Requirements Artifacts of Various Sizes

Parisa Ghazi, Martin Glinz
Department of Informatics, University of Zurich, Switzerland
{ghazi, glinz}@ifi.uzh.ch

***Abstract*—Requirements documentation is essential for developing software systems of non-trivial size. The cost of creating and maintaining documentation artifacts in terms of time and effort is significantly influenced by the tools with which engineers view, navigate and edit documentation artifacts. However, there is not much evidence about how well documentation tools actually support engineers, particularly when dealing with artifacts that are larger than the available display screen and with multiple artifacts at the same time. Therefore, we conducted an exploratory study based on 29 interviews with software practitioners to understand the current practice of presenting and manipulating artifacts in documentation tools, and how practitioners deal with the challenges encountered. Our study shows that a significant number of artifacts cannot be viewed entirely, even on large screens. Moreover, more than half of the participants use four or more artifacts concurrently. Nevertheless, current tools only provide primitive capabilities for handling concurrent and large artifacts, thus forcing engineers to create, for example, mental images of the currently used artifacts or use workarounds such as hanging printouts to the wall. Our results may trigger new research and help improve requirements engineering tools.**

***Index Terms*—Documentation, Artifacts, Requirements engineering, Software engineering tools, Interview, exploratory study**

I. Introduction

In professional software development, poor documentation increases the probability of failure [1]. This is particularly true for requirements, where missing or deficient documentation may lead to developing the wrong product. On the one hand, proper documentation reduces the cost of software development by shortening the duration of tasks and reducing rework [2]. On the other hand, high quality documentation costs: it demands time and effort to create and maintain. This cost is one of the main reasons that documentation is often poor and documents are outdated in software projects [3], [4].

Software tools are largely involved in the creation, maintenance and management of documents, including requirements. Tools support practitioners when creating and editing documents, generating additional information (e.g., summaries, conflicts or traceability links) and presenting the information in a practically useful form [5]. In all these cases, the way how tools present documentation artifacts to users has a profound impact on the user experience when understanding and manipulating artifacts: it may make documentation tasks cost-effective and enjoyable, or cumbersome and frustrating.

To achieve a more profound understanding of the impact of information presentation on software practitioners when manipulating documentation artifacts, we designed and conducted an exploratory study where we interviewed 29 practitioners from eleven countries. Eight out of 29 interviewees can be considered to be requirements engineers (business and software analysts), while the others are architects, developers, testers, and project managers. However, when we analyzed the interview data separately for every role, the results were not significantly different from those obtained for all interviewees (see Sect. III-B). Hence we conclude that our results not only characterize how practitioners deal with large and interconnected documentation artifacts in general, but that these results are equally valid for how requirements engineers deal with requirements artifacts.

Our goal was to (1) examine the size and interconnectivity of documentation artifacts (and how these two factors affect the user interaction with such artifacts), (2) investigate the challenges related to information presentation that practitioners face when interacting with artifacts, and (3) explore what methods practitioners use to overcome the identified challenges and how effective they are.

The outcome of this study will help build tools with better presentation interfaces that allow for effective and convenient manipulation of documentation artifacts.

The remainder of this paper is organized as follows. We define some key terms in Section II. In Section III, we describe our research methodology. Our key findings are presented in Section IV, while Section V summarizes our findings. In Section VI, we explain the threats to validity of our study and how we tried to overcome them. Section VII presents related work. Section VIII concludes with a summary and outlook.

II. Definition of Terms

In this section, we define the terms *artifact*, *screen size* and *'fit on a screen'* that we will use frequently in this paper.

Artifact. In the context of this paper, an *artifact* is any kind of textual or graphical document, with the exception of source code. Artifacts may be, for example, textual requirements documents, graphic models (including UML diagrams), glossaries, charts, or sketches. We excluded source code as we are primarily interested in artifacts relevant to Requirements

Engineering (RE), and the tools used for handling source code are different from tools for handling documentation artifacts.

Screen size. The way of interacting with artifacts not only depends on the artifacts themselves, but also on the *size of the available screen(s)* for presenting and editing the artifacts. Practitioners use screens of various sizes and may also use multiple screens of different sizes simultaneously. When analyzing our data, we use the aggregated screen size or the maximum screen size depending on our analysis purpose. The reason is explained whenever such a decision is taken.

'Fit on a screen'. Depending on the size of an artifact and screen, an artifact may *fit on a screen*, i.e., it can be displayed entirely in a readable and editable size. Otherwise, technical means such as scrolling or zooming are required to view and manipulate an artifact.

III. Research Methodology

To investigate how practitioners interact with artifacts and how they deal with the challenges encountered when interacting with artifacts that do not fit on their screen, we conducted an exploratory study [6] based on semi-structured interviews [7]. So we were able to collect quantitatively analyzable data on the one hand, but, by asking open questions, also obtained information about how practitioners actually deal with the challenges they encounter on the other hand. Consequently, our data set is partially composed of qualitative data [8]. This format also gave us the chance to explain the questions well enough to avoid misunderstandings and collect more accurate answers.

A. Research questions

From the goal of the study, we derived the following research questions.

RQ1. What are the size and interconnectivity of the software and requirements artifacts used in software development industry?

Interacting with artifacts in software tools is more challenging when the artifact has certain properties such as being larger than the screen, or being connected to other artifacts so that the practitioners need to work on multiple artifacts at the same time. Therefore, we decided to investigate the size and interconnectivity of the artifacts first.

RQ2. What challenges do practitioners encounter when working with software and requirements artifacts?

Secondly, we wanted to know the challenges of interacting with artifacts of different size and interconnectivity. This was possible by encouraging the participants to tell us about all the challenges they encounter related to navigating, manipulating, and managing artifacts. Our focus was on challenges that were related to the presentation of artifacts on the screen.

RQ3. What methods do practitioners use to handle the interaction challenges of working with software and requirements artifacts?

Finally, having identified the challenges experienced by practitioners, we study how they try to overcome them, e.g., whether they use special features of software tools or have other workarounds.

B. Study design

1) Initial preparation: Our semi-structured interviews were based on an interview instrument[1], which was first elaborated as a list of questions linked to the RQs and the goal of the study. The interview instrument was designed following the guidelines stated by Oates [9]. Then it was improved in two rounds of evaluation and feedback: first, it was evaluated by a group of RE experts to discover possible ambiguities and shortcomings. Second, we conducted two pilot interviews with a researcher from the University of Zurich and a practitioner.

The interview instrument comprises four sections: (1) characterization of the company and the interviewee, (2) properties and types of artifacts used by the interviewees and the tools they use to handle them, (3) challenges interviewees encounter when working with certain types of artifacts, and (4) how they deal with these challenges.

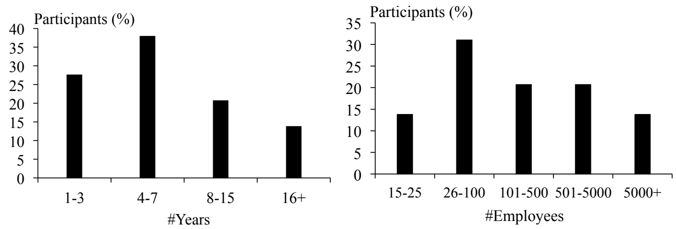
2) Selection of Participants and Demographics: The 29 practitioners we interviewed can be categorized into five roles: we had eight *requirements engineers* (business and software analysts), five *software architects*, nine *software developers*, four *project managers*, and three *software testers*.

Requirements artifacts are used almost in all phases of software development. To obtain a comprehensive view of tools and challenges related to artifact creation, modification, comprehension and management, the study cannot be constrained to a specific phase or role of software development. Otherwise, some of the mentioned aspects will be overlooked.

Many roles in software development work with multiple types of artifacts (e.g., requirements artifacts and design artifacts). Asking participants to consider only one type of artifact when answering the questions would result in inaccurate data, since it is possible that the participants unintentionally consider wrong types for some parts of the questions. Assuming that documentation is done in a similar way in different phases of software development, we neither restricted our study to a particular development role, nor asked the participants to consider specific types of documents. In the analysis phase of our study, we searched for correlations between the roles and other parts of the data set (e.g., size of artifacts and screens, various challenges), but did not find any (see Figure 7 for an example). This indicates that there are no significant differences among the different phases and roles of software development with respect to the questions we are exploring.

We defined two criteria to ensure recruiting suitable participants for our study. We looked for software development practitioners who (1) had at least one year of experience of being a member of a software production team and (2) used software and requirements artifacts (textual and graphical) on a daily basis during their work. The self-evaluation of the participants' experience in working with artifacts is shown in Figure 1a. The largest group of participants (38%) reported between four and seven years of experience in working with requirements and software artifacts.

[1]http://www.ifi.uzh.ch/rerg/people/ghazi/InterviewInstrument2016.pdf



(a) (b)

Fig. 1. Distribution of participants with regard to (a) years of experience and (b) size of the company measured in number of employees

With respect to company size, our study covers a wide range, from under 25 to more than 5000 employees. The largest group of participants works in companies having between 26 and 100 employees. Figure 1b shows the details.

When recruiting participants, we combined two types of sampling: snowball and random sampling [10]. For the snowball sampling, we started from our immediate acquaintances who were active in any phase of software development such as requirements engineering, software design, software development and testing. We sent an e-mail containing a short description of the research and the selection criteria mentioned above, making it clear that participation is voluntary. We asked them to send us a short description of their duties and whether they work with both textual and graphical artifacts or not. After each interview, we asked them to introduce other practitioners who fit the criteria. The majority of the participants believed that another interview with a person from their company would result in similar data. Although the redundant data could help us in validating the data set gathered, we decided to invest our resources on increasing the variety of the participants, and recruit the next group of participants from the acquaintances of the first group working in other companies. The majority of our participants (86%) were recruited by snowball sampling. For the random sampling, we used a social network of professionals (LinkedIn). We sent LinkedIn messages (InMails) to a randomly selected set of practitioners and asked them whether they comply with our selection criteria.

Eventually, we interviewed 29 practitioners from 29 different companies, located in eleven countries from seven geographical regions, as depicted in Figure 2.

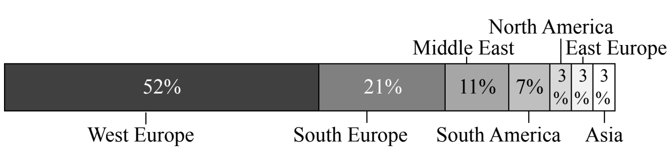


Fig. 2. Geographical distribution of participating companies

3) Data Collection and Analysis: The interviews were conducted between November and December 2015. Their duration varied between 40 and 87 minutes, with an average of 56 minutes. We conducted the interviews over Skype or Google Hangouts, except for three, which were conducted face-to-face. All interviews were recorded and transcribed completely.

We started the data analysis by extracting the quantitative data. Then we summarized the qualitative data without altering or transforming them. Next, we categorized the extracted data and assigned them to the related research questions. At this time we employed a qualitative analysis approach [11] and refined the data by combining the answers with the same meaning expressed in different words. Aggregating and structuring the data helped us reach our findings.

IV. Key Findings

In this section, we discuss the findings that are directly related to our goal and research questions. For each of our three research questions, we present the corresponding findings. For every finding, we provide the evidence from the interviews and a short discussion of the importance of that finding.

A. Properties of Artifacts (RQ1)

It was crucial for us to know the size and interconnectivity of artifacts, so we asked about the size, number and the maximum number of artifacts used simultaneously. In addition we asked about the size and number of screens used by the participants.

Finding FA.1. Only about one third of the graphical artifacts used by the interviewed practitioners fit on their screens.

Evidence for FA.1. We explicitly asked the participants about the percentage of their graphical artifacts that fit on their screen. Figure 3 visualizes this information. In this question we made a distinction between textual and graphical artifacts because textual artifacts have their own way of navigation, search and management. The bounding box represents the entire set of the participant's artifacts. This space is partitioned into 29 vertical bars. Each bar represents a participant and is filled according to the participants' answer. We sorted the participants based on the percentages in order to have the filled areas on one side and not-filled areas on the other side. The total not-filled area in the resulting chart (Figure 3) represents 65% of the artifacts that do not fit on the respective participants' screens.

Discussion. One can argue that the participants' screens have different sizes and the percentages found depend on the actual screen sizes. Although this argument is true, the chart still shows the percentage of the artifacts that are being used on screens that do not permit to view these artifacts entirely. In the next finding we have eliminated this dependency.

Finding FA.2. About forty percent of the graphical artifacts do not fit on the largest screen reported in this study.

Evidence for FA.2. We wanted to investigte the artifact size in a way that does not depend on the participant's screen size. The difficulty was that no common measure for artifact size exists that everybody understands and that allows comparisons. To overcome this problem we made two decisions: (1) We decided to use screen size as the measure for artifact size. Thus, we consider the size of an artifact is the size of the smallest screen that can accommodate the entire artifact in a readable size (without excess effort to read). (2) Since it was nearly impossible to ask participants to describe all of

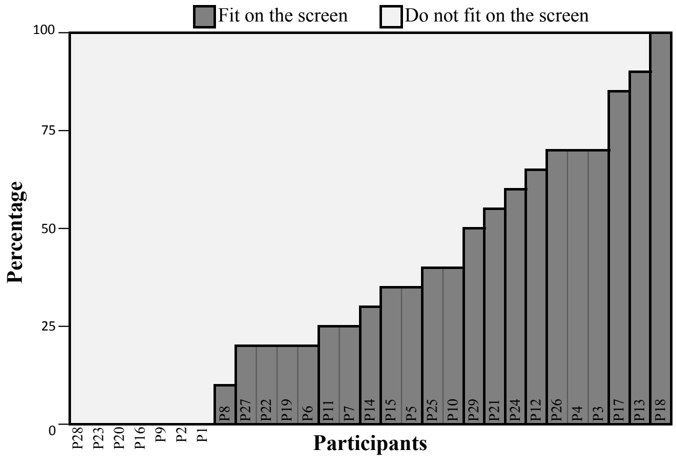


Fig. 3. Percentage of graphical artifacts that fit on the participants' screens

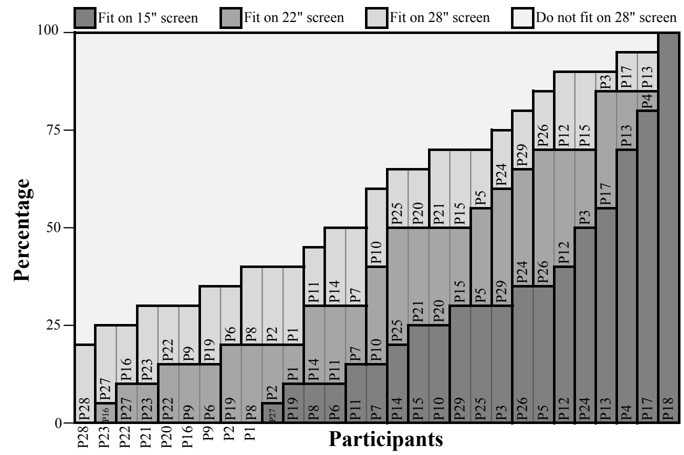


Fig. 5. Percentages of graphical artifacts that fit on different screen sizes

their artifacts with this measure, we asked a simpler question that led us to find the distribution of their artifacts' sizes. In particular, we asked the participants to estimate, in percent, the amount of their artifacts that fit on screens of four different sizes: 11-inch, 15-inch, 22-inch, and 28-inch. This question not only is simpler, but provides more information.

To use the most accurate data we had, we referred to the frequency of screen sizes illustrated in Figure 4: the 22-inch screen is the most used, followed by the 15-inch screen. In addition to these two screen sizes, we used the percentages provided for 28-inch screens to include the maximum percentage of the artifacts that fit on the largest monitor (according to this research). The total number of screens is higher than the number of participants because many of them had more than one screen. Figure 5 shows the result in the same format as FA.1 except that here we show the percentages of graphical artifacts fitting on three different screen sizes.

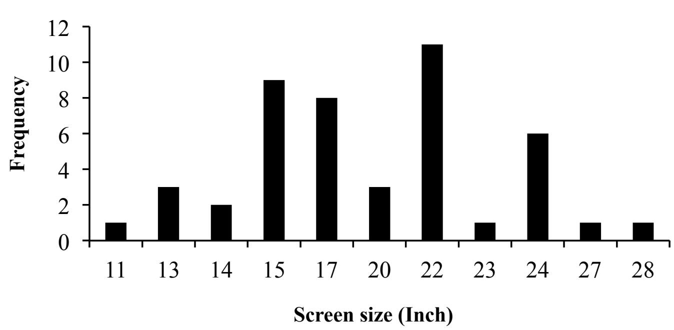


Fig. 4. Distribution of screen sizes of the participants

According to our finding, 24% of the participants' artifacts fit on a 15-inch screen, 42% of the participants' artifacts fit on a 22-inch screen, and 58% of the participants' artifacts fit on a 28-inch screen.

Discussion. In Figure 5, the part with the lightest gray ("Do not fit on 28" screen") has taken more than 41% of the area and represents the artifacts that do not fit even on the largest screen reported in this study. Consequently, supplying larger screens will not solve the problem completely: alternative solutions are needed.

Finding FA.3. More than half of the interviewed practitioners use four or more artifacts at the same time.

Evidence for FA.3. We asked the participants about the maximum number of artifacts they use simultaneously. We received different answers ranging from one to twenty. The box plot in Figure 6 shows the distribution of the answers. Half of the participants use between three and six artifacts at the same time; the median is four. Two practitioners with twelve and twenty artifacts used at the same time are outliers, therefore we can say that the number of artifacts being used concurrently ranges between one and ten.

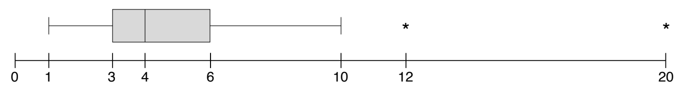


Fig. 6. Number of artifacts used at the same time

Discussion. This finding emphasizes the importance of screen space and how it is used to present artifacts. From the fact that a significant portion of artifacts are larger than the available screens (from FA.1 and FA.2) and more than half of the practitioners use four or more artifacts at the same time, we conclude that practitioners either need larger screens (which is limited by cost and technology) or the existing screen space must be used in a smarter way when presenting artifacts to their users.

B. Challenges in Working with Large Artifacts and Multiple Artifacts at the Same Time (RQ2)

After asking the participants about the properties of their artifacts, we asked about the challenges they experienced when working with artifacts larger than their screens, and the challenges they face when working with multiple artifacts at the same time.

Finding FB.1. "Relying on memory", "Searching for information", and "Maintaining the overview" are the most important challenges in handling large artifacts.

Evidence for FB.1. After gaining a perspective of the participants' artifact size and screen size, we asked them about the challenges of working with artifacts that are larger than the available screen. After gathering all challenges, we first created a comprehensive list of them. To guarantee atomic and concrete challenges, we remove general ones (e.g., "*Working with large artifacts is not efficient*") and dependent ones (e.g., "*This type of artifact takes so much time*"). Afterwards we grouped similar challenges that were expressed in different words. For example, below we give some quotes about how participants rely on their memory. P23: "*It increases your cognitive overhead because you do not remember where things are*", P10: "*You have to imagine what is located in the part of the picture you cannot see*" and P14: "*Because I forget things easily I have to take notes in another place*". Table I presents the list of challenges and their frequency (number of participants mentioning that challenge). The calculation of the priorities is explained below.

TABLE I
CHALLENGES OF WORKING WITH ARTIFACTS THAT ARE LARGER THAN THE AVAILABLE SCREEN

ID	Challenge	Priority	# Participants
C1.1	Relying on memory	1	18
C1.2	Searching for information	2	2
C1.3	Maintaining the overview	3	20
C1.4	Too much scrolling and zooming	4	29
C1.5	Knowing the current location	5	5
C1.6	Following the links	6	7

The frequency of mentioning a challenge alone is inadequate to show the importance of the challenge, as different participants may be affected by the challenges to different extents. Therefore we decided to rank the participants based on how heavily they are affected by the challenges of working with artifacts that are larger than their screen(s) and to use this ranking for prioritizing the challenges listed in Table I. For this purpose we assumed that people who have larger artifacts and smaller screens are more challenged than others.

As a first step, we computed the average artifact size of each participant using the Cumulative Distribution Function (CDF). We gathered three points for the CDF of the artifact size for each participant: the participant's estimation of which percentage of artifacts fit on 15-inch, 22-inch, and 28-inch screens, respectively (according to FA.2). CDF is calculated by the following formula [12]:

$$CDF(x) = \frac{1}{2}\left[1 + erf\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)\right] \tag{1}$$

x represents the screen size and and $CDF(x)$ is the percentage of artifacts that fit on a screen of that size. These values are known from the interviews. μ and σ are mean and variance respectively. erf is called error function and is already known [12]. So we rewrite the formula as:

$$\sqrt{2}\, erf^{-1}\left(2\, CDF\left(x\right) - 1\right)\sigma + \mu = x \tag{2}$$

This is a linear equation with regard to the parameters σ and μ. Therefore, by plugging the mentioned three points, we can calculate σ and μ. In the rest of this paper, the calculated μ is called S_μ. Since it is the mean of the artifact sizes for each participant, it indicates "the size of the smallest screen that can accommodate half of the artifacts in a readable size". Figure 7 shows the overall and role-wise distribution of S_μ.

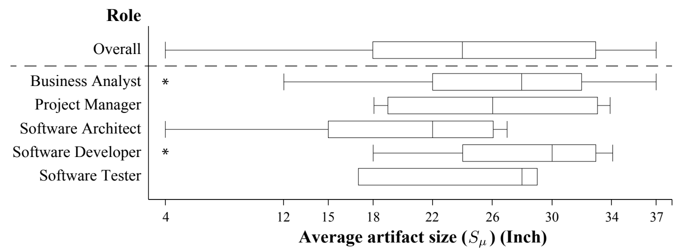


Fig. 7. Overall and role-wise distribution of the average artifact size (S_μ)

Having calculated S_μ for all participants, we now can rank them with respect to the degree that they are affected by the challenges of working with artifacts that are larger than the available screen(s). We do this by assigning a point to each participant in a coordinate system with screen size as x-axis and S_μ as y-axis, and calculating the distance of each point from a point which represents a hypothetical person who is more under influence of these challenges than all others in our dataset. This hypothetical person has a screen size of 12 (less than everyone else) and a S_μ value of 40 (more than everyone else). The result is depicted in Figure 8. The closer a participant is to the hypothetical person in the top left edge, the more she or he is affected by the challenges of working with artifacts that are larger than the available screen(s). The rank of participants who have more than one screen was calculated based on their largest screen, assuming that they work with large artifacts on their largest screen.

We sorted the participants based on the calculated distances and ranked them so that the participant with the lowest

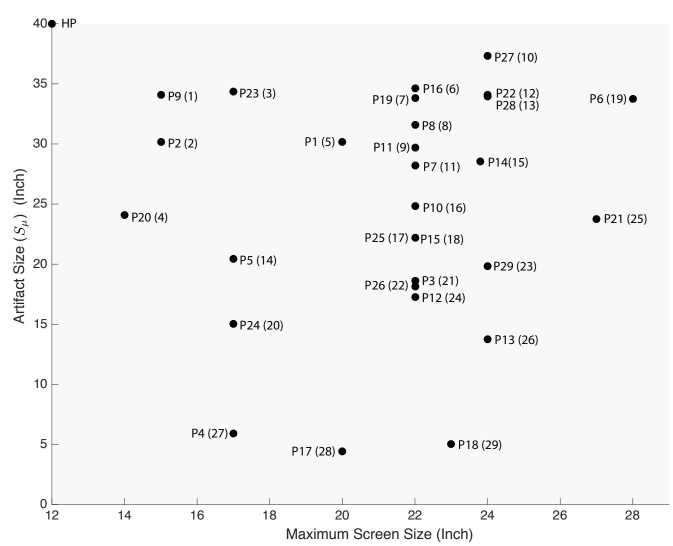


Fig. 8. Ranking of the participants with respect to artifact and screen size used. The labels depict the number and rank of the participants. HP denotes the hypothetical person in the top left edge, from which the distances are measured.

distance has the highest rank of 1 and the participant with the highest distance has the lowest rank of 29. Then we propagated the ranks of the participants to the challenges they mentioned. For example, if X is a challenge mentioned by three participants with ranks (1, 8, 18), we ranked X with the average of the participant ranks, which is nine. Finally, we prioritized the challenges according to their ranks as shown in Table I.

Discussion. The most frequently mentioned challenge is the problem of too much scrolling and zooming (C1.4), which is an obvious impact of working with an artifact that is larger than the screen. However, after prioritization, the top challenge is the participants' need to rely on their memory (C1.1), particularly to avoid excessive scrolling and zooming (C1.4). P27: "*I use my memory if I can avoid zooming and scrolling around*". Maintaining the overview over the artifact (C1.3) is the third top challenge with respect to the number of participants who mentioned this challenge. Interestingly, searching for information (C1.2) is the second ranked challenge, although it was mentioned by two participants only. P3: "*When you want to find information in an artifact and the artifact is a big one, it is very hard. Searching information in larger artifacts is harder*". This illustrates to what extent searching for information can be cumbersome when having large artifacts and a small screen.

Finding FB.2. "Switching between artifacts" and "Working in too small windows" are the most important challenges when working with multiple artifacts.

Evidence for FB.2. The challenges of working with multiple artifacts are identified and prioritized similarly to FB.1. In this case, we identified three related parameters for ranking the participants: screen size, S_μ (smallest monitor size that accommodates half of the artifacts) and the number of artifacts used at the same time. We assigned a point in the three-dimensional space to each participant. We assumed that a person with large artifacts, a small screen and a high number of concurrently used artifacts is stronger impacted by the challenges of working with multiple artifacts than others. Therefore, we calculated the distance between each participant point and the point showing the hypothetical person having an extremely small screen, a S_μ value of 40 (higher than everyone else), and using 20 artifacts at the same time (the highest in our dataset). When analyzing, we aggregated the screen sizes of participants who use multiple screens, since they can open different artifacts on different screens at the same time and use all of the available screen space. The result is depicted in Figure 9.

We used the calculated distances to rank the participants. Then we propagated the ranks of the participants to the challenges they mentioned and sorted the challenges based on the average of their ranks. Table II shows the result.

Discussion. Practitioners mostly use multiple screens and multiple windows to work with multiple artifacts at the same time. According to the participants, the most cumbersome task is switching between windows (C2.1), and it gets worse when the tools only support multiple tabs (instead of multiple

TABLE II
CHALLENGES OF WORKING WITH MULTIPLE ARTIFACTS AT THE SAME TIME

ID	Challenge	Priority	#Participants
C2.1	Switching between artifacts	1	23
C2.2	Working in too small window	2	12
C2.3	Changing focus	3	8
C2.4	Knowing the relations between artifacts	4	6
C2.5	Finding the right window	5	16
C2.6	Arranging windows	7	4
C2.7	Memorizing	8	10
C2.8	Finding the current focus position	9	2

windows). To make navigation between artifacts easier, some of the practitioners arrange windows side by side. This results in easier switching, but raises two other challenges: (1) Each artifact has less dedicated space, therefore the user has to work in a smaller window (C2.2). P16: "*Most of the time working with windows side by side is really difficult because there is less space to work*". (2) The windows need arrangement (C2.6). P22: "*I arrange the windows regularly and resize them. When you expand or close a window you have to move the other windows*". When moving back and forth between multiple artifacts, practitioners may forget the exact location where they left an artifact and need some time to find the right location when they return. Therefore changing focus causes interruption in their work (C2.3). Finding the window that they need in a particular moment is a challenge (C2.5), since windows tend to hide under each other and using keyboard shortcuts to move through windows consecutively takes time and is error-prone. Recent operating systems (e.g., OS X) provide an overview of the open windows, but this becomes increasingly useless when a large number of windows is open. Morover, bringing up the overview still takes time and does not work with multitab systems. When using multiple windows,

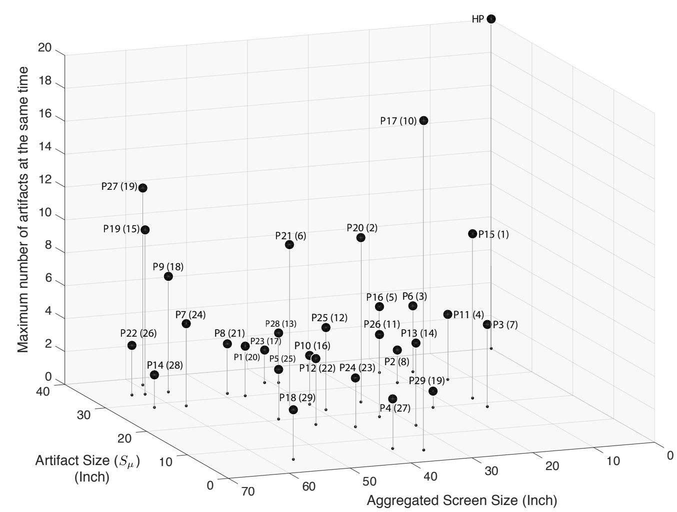


Fig. 9. Ranking of the participants with respect to artifact size, screen size and number of artifacts used. The labels depict the number and rank of the participants. HP denotes the hypothetical person, from which the distances are measured in the three-dimensional space. Note that the actual three-dimensional distances of the points are different from what they seem to be in the projection used in this figure.

there is no information about the relationship between the artifacts inside the open windows (C2.4). Since the space for each artifact is smaller, more information is located outside of the screen. Therefore, the users have to keep more information in their mind (C2.7). The location of the cursor is easily mistaken when multiple windows are open (C2.8).

C. Dealing with the Challenges (RQ3)

We asked the participants how they deal with the challenges of handling artifacts. In particular, we were interested in knowing what methods they use to overcome the challenges mentioned above, what features their tools provide, and what they do when the tools do not support them sufficiently.

Finding FC.1. Practitioners use their memory extensively.

Evidence for FC.1. One of the main challenges in working with large artifacts, according to FB.1 and FB.2, is "relying on memory". To know how many of the participants use their memory intentionally, we asked them directly whether they memorize any part of an artifact to use it elsewhere or not. 69% of the participants answered affirmatively. We asked the participants who responded "no" how they handle the situations where they need a piece of information from another artifact or another part of the current artifact which is not on the screen at the moment. We found that 21% of the participants use copy and paste functions instead of keeping information in memory. They also mentioned that this method is not applicable to graphical information easily and they sometimes have to take screenshots. P19: "*I don't memorize. Instead I use copy-paste. If it is a diagram, I would make a screenshot of it*". Finally, only 10% of the participants answered "no" decisively.

We asked the participants if better visual memory positively influences their performance. 82% of the participants admitted that better visual memory affects their performance positively in working with graphical artifacts.

Discussion. The answers to these two questions prove that practitioners rely on their memory extensively when working with artifacts and show that they compensate their inadequate memory power by using copy-pasting and taking screenshots, which is error-prone and time-consuming in turn.

Finding FC.2. Traditional zooming and scrolling are the dominating techniques for handling large artifacts.

Evidence for finding FC.2. As stated in FA.1 and FA.2, practitioners often work with artifacts that are larger than their screens. Many artifacts even do not fit on the largest screens reported in this study (FA.2). However, when we asked them how they handle such artifacts and what features tools provide for this purpose, we found that they mostly use simple traditional methods such as scrolling and zooming.

Cockburn [13] categorized visualization techniques that help handling larger-than-screen artifacts into four classes: zooming, overview+detail, focus+context and cue based techniques. We asked our participants how they handle large artifacts to know the techniques implemented in commercial tools. We found that traditional zooming and scrolling are the most basic techniques used for this purpose. In addition to zooming and scrolling, only three participants use tools that provide an overview+detail feature. The applications they use show an overview of the artifact in a small window and they can navigate inside the artifact by using this small overview. None of the interviewees have any focus+context or cue-based techniques available. Obviously the features that exist but are not known by the participants are not counted in this report. Three interviewees explained that they avoid having large artifacts by defining different layers of abstraction. The result is visualized in Figure 10.

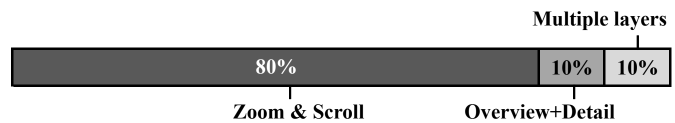


Fig. 10. How participants handle artifacts that are larger than the screen

Discussion. By maintaining multiple layers of abstraction, some of the participants could manage to have smaller artifacts at the cost of increasing the number of artifacts and having redundant data in multiple artifacts. Managing a larger number of artifacts with redundant data needs additional effort. Working with multiple artifacts gives rise to other issues that we discussed in FB.2. In this regard, participant 18 describes his needs as "*In my tool, different diagrams which show different layers of abstraction cannot be interconnected. What I really like is to start on a high level and go to a really detailed level, and get to the other diagrams that show the layers beneath. I do not know any software that has this kind of zooming*".

Finding FC.3. Almost all practitioners need to have an overview of the artifacts.

Evidence for FC.3. 65% of the participants keep an overview of the artifact in their mind and 32% of the participants employ other techniques to maintain an overview of the artifact (see Table III). Only one participant mentioned that he does not need to have an overview.

TABLE III
ALTERNATIVE TECHNIQUES TO MAINTAIN OVERVIEW

ID	Alternative technique to maintain overview
P2	Zooming out
P3, P16	Printing and hanging the artifact on the wall
P12	Taking notes
P15	Opening the artifact twice
P17, P18, P20	Creating a higher abstraction level of the artifacts
P23	Using the overview provided by the application

Discussion. The list of techniques given in Table III reveals that participants primarily use simple techniques (such as zooming out or opening an artifact twice) or workarounds (such as printing or taking notes). The only sophisticated and systematic technique employed is creating a higher abstraction level. However, this is complicated and time-consuming as the participants have to do this manually.

Finding FC.4. Non-software approaches are mostly used for handling the challenges of working with multiple artifacts at the same time.

Evidence for FC.4. To handle the challenges related to the number of artifacts used simultaneously, 79% of the participants use multiple screens (two or three screens) and 24% of them print some of their artifacts in addition to having multiple screens (Figure 11).

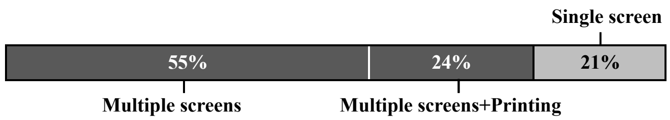


Fig. 11. How participants handle multiple artifacts at the same time

Discussion. Using multiple screens and printing increases the number of artifacts that practitioners can view simultaneously without any switching. The fact that the majority of practitioners uses multiple screens demonstrates their need for concurrently working with more than one artifact. However, both multiple screens and printing have drawbacks and limitations. The main drawback is that both exacerbate the challenge of repetitive change of focus, which is one of the main challenges we found (FB.2). Moreover, the number of concurrently usable screens and printouts is limited in three dimensions: (1) cost, (2) available space to place screens and printouts in a work environment, and (3) at some point searching for the needed information in a multi-srcreen and multi-printout environment becomes as difficult and cumbersome as keeping the same information in multiple windows on a single screen.

V. Summary of Findings

The objective of this research was to portray (1) how size and interconnectivity of artifacts affect the performance of practitioners when interacting with artifacts, (2) how practitioners deal with the challenges encountered when working with multiple artifacts concurrently as well as with artifacts that are larger than the available screen, (3) how software tools address these challenges.

In essence, we found that software artifacts are large (FA.1, FA.2), are used concurrently (FA.3), and while the practitioners face various challenges (FB.1, FB.2) in working with these artifacts, the software tools do not provide rich functionalities to help them (FC.2, FC.4). Therefore, practitioners have to use their memory or primitive functionalities such as copy-pasting (FC.1), which are time-consuming and error-prone. We found that the interviewed practitioners use up to three screens to have more space and to handle the challenges of working with multiple artifacts (FC.4). They print an overview of their interconnected artifacts to avoid getting lost (FC.3). All these findings are summarized in Table IV, grouped by the research questions defined in Section III.

TABLE IV
Key Findings

RQ1: Properties of the Artifacts
FA.1. Only about one third of the graphical artifacts used by the interviewed practitioners fit on their screens.
FA.2. About forty percent of the graphical artifacts do not fit on the largest screen reported in this study.
FA.3. More than half of the interviewed practitioners use four or more artifacts at the same time.
RQ2: Challenges in Working with Large Artifacts and Multiple Artifacts at the Same Time
FB.1. “Relying on memory”, “Searching for information”, and “Maintaining the overview” are the most important challenges in handling large artifacts.
FB.2. “Switching between artifacts” and “Working in too small windows” are the most important challenges when working with multiple artifacts.
RQ3: Dealing with the Challenges
FC.1. Practitioners use their memory extensively.
FC.2. Traditional zooming and scrolling are the dominating techniques for handling large artifacts.
FC.3. Almost all practitioners need to have an overview of the artifacts.
FC.4. Non-software approaches are mostly used for handling the challenges of working with multiple artifacts at the same time.

VI. Threats to Validity

When a large part of a study is a qualitative exploratory research, it is not possible to resolve all threats to validity. However, we tried to minimize their effect on the final result. Below we discuss the usual four categories of validity [14].

Conclusion validity refers to finding a relation between data if it exists. Measurement reliability can affect conclusion validity. Therefore, to make our measures clear for the participants, we described our measures in detail and in a step-by-step manner. In addition, all interviews were conducted by the first author. When we needed a new measure, we defined it by combining other well-known measures (e.g., screen size and artifact size). We verified the consistency of our measures by asking duplicate questions. Moreover, we discussed the questions with RE experts and did a pilot study to avoid misunderstandings. Since we asked the participants to imagine having a screen size that they did not have in reality, still the accuracy of the gathered data depends on how accurately they can imagine that situation. Also, we made assumptions about what kind of people are more likely to have experienced challenges. Different assumptions might have resulted in different orders in our lists. Furthermore, we tried to hold the interviews under similar conditions by scheduling the meetings in advance and asking the participants to be in a non-disruptive enviroment.

Internal validity of an interview refers to making sure that the differences in the answers received are only because of the known differences among participants. Questions remained the same during the whole duration of the study. All 29 interviews were performed within a relatively short period of two months to avoid any software or hardware technology advancement. All participants were self-motivated and we did not offer any compensation.

Construct validity ensures that questions actually ask what they are supposed to ask. For example, we cannot guarantee that the participants remember everything related to our questions during the interview. So if a participant did not mention a challenge, this does not necessarily mean that they did not face that challenge. Therefore, in the analysis phase, we tried to minimize the influence of the frequency of the answers by prioritizing the challenges based on other factors. Moreover, although we did not have any hypothesis or expectation about the results, we were careful not to let the participants guess any hypothesis or expectation by mistake. In addition, to increase the accuracy of the measurement and make the participants more relaxed, we informed them that the data will be used and presented anonymously. We gathered information from various sources to avoid mono-operation bias.

External validity of a research means that the results are generalizable. For this purpose, the selected sample (the interviewees in our case) should not have certain features in common. This is very hard to achieve in an interview-based study. Some features were inevitably shared by all participants such as being volunteers who are interested in contributing to a scientific study and are social enough to answer our e-mail and participate in a one-hour long interview. To avoid bias, we defined our criteria for selecting participants as simple as possible and used two different types of sampling. The variety of our final sample in terms of country, roles and company size shows that we were successful. Nevertheless, we cannot claim that our sample of 29 practitioners is statistically representative for the whole software development community.

VII. Related Work

Screen space for displaying information has always been limited. Therefore, the way users interact with the information displayed on the screen has been the focus of a large number of studies. These studies aimed especially at optimizing the information presentation. Two different approaches can be taken for this purpose: increasing the screen space or utilizing the available screen space more efficiently. In the first approach, using large screens or arrays of multiple screens is being investigated. Czerwinski et al. [15] conducted an empirical study to examine the productivity benefits of larger display screens and found a significant performance advantage. Lischke et al. [16] used multiple monitors to have a wall-size screen in an empirical study, and measured the task completion time in different settings. They reported that the optimal monitor number is three. In the second approach, techniques such as zooming, overview+detail, focus+context and cue-based methods are employed to display as much useful information as possible on the screen [13]. Lam et al. [17] analyzed 22 studies that implemented such techniques to extract design guidelines indicating when and how each of these techniques should be used.

However, both of these approaches give rise to other challenges such as arranging the windows and tracking the mouse pointer [18]. Therefore, for improving the performance of existing user interfaces, it is not sufficient to just increase the screen size or employ a smart visualization mechanism. Instead, the design of user interfaces needs adaptation, which requires understanding the new challenges. Furthermore, the findings of these researchers depend on the information type (e.g., graph or 3D model), the interaction type (e.g., comprehension or manipulation) and the users. Consequently, the provided guidelines need to be tailored to requirements engineering.

Although many studies evaluated and compared the usability and performance of user interfaces of different tools, only few of them discussed requirements engineering tools. For example, in an exploratory study on three different software exploration tools, De Alwis et al. [19] found no evidence of any practical benefit of using the selected tools. Roehm et al. [20] found that developers do not know some standard features of their tools. In spite of many existing specialized tools, Forward et al. [21] reported that most preferred tools for documentation include word processors and text editors. These reports mostly demonstrate the ineffectiveness of the tools. One of the reason is that the real interaction challenges of practitioners are not well identified in the first place.

Requirements visualization is another broad area of research that investigates how graphical models of requirements should be created [22]. However, the research in this area does not address how the created graphical models should be presented to users. For example, Cleland-Huang et al. [23] proposed visualization techniques such as hierarchy structures to enhance the understandability of artifacts in automatic tracing tools. Reddivari et al. [24] designed a tool to support the exploration of requirements via quantitative visualizations. The true benefits of these tools will not be realized unless the artifacts are presented to users in the most effective form. For instance, Reinhard et al. [25] [26] developed a custom-made presentation technique to fully exploit the potential of the requirements modeling language ADORA.

In addition to enhancing the understandability of the artifacts, software engineering tools can provide cognitive support [27]. If a complex diagram is not presented hierarchically, viewers have to derive the hierarchy in their mind [28]. Cornelissen et al. [29] established a set of metrics for scenario diagrams to recommend a number of abstractions that should be used to have the desired amount of detail. Bennett et al. [30] reported the usefulness of their interaction features for sequence diagram navigation. Presenting information in a clear pattern helps remembering the relationships [28]. In a data-intensive field like requirements engineering, offloading some of the cognitive load is a requirement for any tool which supports viewing and editing of artifacts.

Requirements engineers should be able to benefit from all these technologies. However, the existing work related to visualizing artifacts and utilizing screen space efficiently and effectively does not thoroughly investigate the interaction challenges encountered by practitioners. As a result, there is a lack of understanding how practitioners handle large-size artifacts as well as interconnected artifacts that are used simultaneously. Our study contributes to closing this knowledge gap.

VIII. Conclusion and Future Work

In this paper, we presented an exploratory study on characterizing requirements and software development artifacts with regard to their size and interconnectivity, and the challenges that are related to these properties. Our goal was to gain in-depth understanding of the state of practice in this area. To achieve this goal, we interviewed 29 participants from different companies located in eleven countries. Our findings clarify the relation between the mentioned properties of artifacts, the challenges related to information presentation, and how these challenges are handled in reality. We found that practitioners work with artifacts that are larger than their screens and interconnected artifacts that have to be accessed simultaneously. Since the existing software tools do not provide sufficient support and features for conveniently handling such artifacts, our respondents try to address the challenges encountered in various ways. For instance, they heavily rely on their memory or use other methods that are inefficient and most often error-prone (e.g., taking screenshots).

Our findings suggest that improving the user interface of the tools used when working with artifacts would be profitable. The results of this study can guide user interface designers to know the important requirements and how they should present information on the screen when developing software documentation tools. More efficient interaction with artifacts will eventually result in lower time and effort needed for documentation, and finally in lower cost and higher quality.

As a next step, we plan to use the findings of this study to develop a tool-supported approach that enables practitioners to handle large artifacts as well as concurrently used sets of interconnected artifacts in an efficient and effective way. With this tool, we aim at addressing the challenges currently faced by practitioners and reducing the errors that emerge from the ad-hoc methods they use at the moment.

IX. Acknowledgement

We would like to thank all the participants and their companies for their time and valuable contributions that made the data collection possible for this research.

References

[1] I. Sommerville, "Software documentation," *Software Engineering*, vol. 2, pp. 143–154, 2001.
[2] J. Zhi, V. Garousi-Yusifoğlu, B. Sun, G. Garousi, S. M. Shahnewaz, and G. Ruhe, "Cost, benefits and quality of software development documentation: A systematic mapping," *Journal of Systems and Software*, vol. 99, pp. 175–198, 2015.
[3] S. C. B. de Souza, N. Anquetil, and K. M. de Oliveira, "A study of the documentation essential to software maintenance," in *Proceedings of the 23rd Annual International Conference on Design of Communication: Documenting & Designing for Pervasive Information*. ACM, 2005, pp. 68–75.
[4] E. Braun, D. Amyot, and T. C. Lethbridge, "Generating software documentation in use case maps from filtered execution traces," in *SDL 2015: Model-Driven Engineering for Smart Cities*. Springer, 2015, pp. 177–192.
[5] J. M. C. De Gea, J. Nicolás, J. L. F. Alemán, A. Toval, C. Ebert, and A. Vizcaíno, "Requirements engineering tools: Capabilities, survey and assessment," *Information and Software Technology*, vol. 54, no. 10, pp. 1142–1157, 2012.
[6] C. B. Seaman, "Qualitative methods," in *in Guide to advanced empirical software engineering*. Springer, 2008, pp. 35–62.
[7] N. King and C. Horrocks, *Interviews in qualitative research*. Sage, 2010.
[8] C. Robson, *Real World Research*, 2nd ed. Blackwell Publishing. Malden, 2002.
[9] B. J. Oates, *Researching Information Systems and Computing*. Sage, 2005.
[10] F. Shull, J. Singer, and D. I. Sjøberg, *Guide to advanced empirical software engineering*. Springer, 2008.
[11] M. B. Miles and A. M. Huberman, *Qualitative Data Analysis: An Expanded Sourcebook*. Sage, 1994.
[12] W. L. Martinez and A. R. Martinez, *Computational statistics handbook with MATLAB*. CRC press, 2007.
[13] A. Cockburn, A. Karlson, and B. B. Bederson, "A review of overview+detail, zooming, and focus+context interfaces," *ACM Computing Surveys (CSUR)*, vol. 41, no. 1, pp. 2:1–2:31, 2009.
[14] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in Software Engineering: An Introduction*. Kluwer Academic Publishers, 2000.
[15] M. Czerwinski, G. Smith, T. Regan, B. Meyers, G. Robertson, and G. Starkweather, "Toward characterizing the productivity benefits of very large displays," in *Proceedings of INTERACT*, vol. 3, 2003, pp. 9–16.
[16] L. Lischke, S. Mayer, K. Wolf, N. Henze, A. Schmidt, S. Leifert, and H. Reiterer, "Using space: Effect of display size on users' search performance," in *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '15)*. ACM, 2015, pp. 1845–1850.
[17] H. Lam and T. Munzner, "A guide to visual multi-level interface design from synthesis of empirical study evidence," *Synthesis Lectures on Visualization*, vol. 1, no. 1, pp. 1–117, 2010.
[18] G. Robertson, M. Czerwinski, P. Baudisch, B. Meyers, D. Robbins, G. Smith, and D. Tan, "The large-display user experience," in *Computer Graphics and Applications*, vol. 25, no. 4. IEEE, 2005, pp. 44–51.
[19] B. De Alwis, G. C. Murphy, and M. P. Robillard, "A comparative study of three program exploration tools," in *15th IEEE International Conference on Program Comprehension (ICPC '07)*. IEEE, 2007, pp. 103–112.
[20] T. Roehm, R. Tiarks, R. Koschke, and W. Maalej, "How do professional developers comprehend software?" in *Proceedings of the 34th International Conference on Software Engineering*. IEEE Press, 2012, pp. 255–265.
[21] A. Forward and T. C. Lethbridge, "The relevance of software documentation, tools and technologies: A survey," in *Proceedings of the 2002 ACM Symposium on Document Engineering (DocEng '02)*. ACM, 2002, pp. 26–33.
[22] J. R. Cooper Jr, S.-W. Lee, R. A. Gandhi, and O. Gotel, "Requirements engineering visualization: a survey on the state-of-the-art," in *Fourth International Workshop on Requirements Engineering Visualization (REV)*. IEEE, 2009, pp. 46–55.
[23] J. Cleland-Huang and R. Habrat, "Visual support in automated tracing," in *Second International Workshop on Requirements Engineering Visualization (REV 2007)*. IEEE, 2007.
[24] S. Reddivari, Z. Chen, and N. Niu, "Recvisu: A tool for clustering-based visual exploration of requirements," in *20th IEEE International Requirements Engineering Conference (RE '12)*. IEEE Computer Society, 2012, pp. 327–328.
[25] T. Reinhard, S. Meier, and M. Glinz, "An improved fisheye zoom algorithm for visualizing and editing hierarchical models," in *Second International Workshop on Requirements Engineering Visualization (REV 2007)*. IEEE, 2007.
[26] T. Reinhard, S. Meier, R. Stoiber, C. Cramer, and M. Glinz, "Tool support for the navigation in graphical models," in *30th International Conference on Software Engineering (ICSE'08)*, 2008, pp. 823–826.
[27] P. Parsons, "Cognitive activity support tools: Design of the visual interface," Ph.D. dissertation, The University of Western Ontario, 2013.
[28] H. Koning, C. Dormann, and H. van Vliet, "Practical guidelines for the readability of IT-architecture diagrams," in *Proceedings of the 20th Annual International Conference on Computer Documentation (SIGDOC '02)*. ACM, 2002, pp. 90–99.
[29] B. Cornelissen, A. Van Deursen, L. Moonen, and A. Zaidman, "Visualizing testsuites to aid in software understanding," in *11th European Conference on Software Maintenance and Reengineering (CSMR'07)*. IEEE, 2007, pp. 213–222.
[30] C. Bennett, D. Myers, M.-A. Storey, D. M. German, D. Ouellet, M. Salois, and P. Charland, "A survey and evaluation of tool features for understanding reverse-engineered sequence diagrams," *Journal of Software Maintenance and Evolution: Research and Practice*, vol. 20, no. 4, pp. 291–315, 2008.